

Building an Active Atlas for the Mouse Brain

Yuncong Chen¹, Partha Mitra, Harvey Karten², David Kleinfeld², Yoav Freund¹

¹ UCSD Computer Science and Engineering Department

² UCSD Physics Department

Abstract We propose a semi-supervised system for generating a reference atlas from histology image series of the mouse brainstem. The atlas contains the mean and variance of the positions and shapes of nine anatomical landmarks. By learning landmark classifiers, the system registers new data to the atlas. Registered data further improve the generalizability of the classifiers.

Keywords: landmark detection, atlas generation, mouse brain, registration, automated annotation

1 Overview

One of the most important tools to neuroscientists, the brain atlas provides a standard model for the anatomy of a certain species' brain. It helps one identify the location of anatomical structures in a new sample. It also allows one to map different samples to a common coordinate system, so that one can quantify the spatial variance of a particular structure between individuals, or correlate gene expression patterns spatially to identify neural circuitry.

In this paper we focus on building an atlas for the mouse brainstem using light microscopy stained with Nissl. Existing mouse brain atlases, such as the Paxinos' [12], Waxholm Space [6] and the Allen Reference Atlas [4], have some limitations. [YF: In american technical prose, you start with the most significant difference and follow it with the less important ones. I think that the manual work and the lack of registration tools are the most significant differences. Then the variance, and then the specific problem with the brain-stem having less grey-level variation.]

First, a lot of manual work is involved in specifying registration landmarks and making annotations. Second, they do not characterize the variation in the location and shape of structures.

First, they do not have the desired level of details in the brainstem. Second, they are merely visual references. There are no tools for registering new data to the atlas. Third, they are hard to update and refine as new data is collected.

We propose a new type of atlas which we call an "active atlas". On the one hand, an active atlas is an anatomical reference model. It contains the mean and variance of position and shape of major anatomical structures. On the other hand, an active atlas includes landmark detectors. These detectors are texture classifiers trained in a semi-supervised fashion to detect regions with salient texture, such as nuclei and tracts, that can serve as registration landmarks. Such an atlas is "active" because the detectors allow our system to actively detect landmarks on new data for registration, instead

of being a “passive” registration template; and also because these detectors can update themselves from new data, thus not “static”.

This paper proposes a machine-learning based method for developing an active atlas, based on image series of Nissl-stained cryo-sectioned brains.

The rest of the paper is organized as follows. In Section 2 we describe related work. In Section 3 we describe the semi-supervised method we used to create landmark detectors. In section 4 we describe the alignment algorithm and the way in which we quantify the variation in the location of the landmarks. In section 5 we describe our experimental result. We conclude in section 6 with lessons learned and future plans.

2 Related work

[YC: this section needs revising]

Existing atlases of the mouse brain include Paxinos, Waxholm Space [6], and the more recent Allen Reference Atlas [4] and its successors, on top of which gene expression data [8, 10] as well as connectivity information [11] are mapped. The latest version of their common coordinate framework [1] is generated by deformably averaging two-photon microscopy images and manually demarcating structures with the help of immunohistochemistry data and genetic labeling data. Atlases for smaller organisms such as fruit flies [3, 13] and zebrafish [14, 15] are mostly based on volumetric images acquired by confocal microscopy. For registration, the BrainAligner program [13] relies on matching high-curvature points, and the ViBE-Z project [15] used trained classifiers to detect predefined 3D landmarks.

3 Methodology

The pipeline Figure 1

3.1 Learning Texture Classifiers using Convolutional Neural Network

To bootstrap the atlas generation process, an arbitrary stack was picked, and an expert anatomist annotated every other image for major landmarks by drawing polygons on the images (Figure 2). The results are 100 polygons for 19 different landmarks that span the entire brainstem, including nuclei (e.g. facial motor nucleus, trigeminal nucleus) and fiber tracts (e.g. trigeminal spinal tract, superior cerebellar peduncle). Patches of size 224-by-224 that are completely inside of each annotation polygon are extracted. Random patches from outside of any polygon are also extracted as examples of “background”. These patches form the training set for the texture classifier. A multi-way classifier is obtained by fine-tuning the inception-BN neural network [1] using the training set. The network is pre-trained for ILSVRC2012 dataset.

For an input patch, the classifier outputs a 20-dimensional probability vector, indicating the probability that the patch belongs to either the background or each of the 19 landmarks.

The classifier is applied to images in all other stacks. To save computation, instead of predicting for patches centered at every pixel, the classifier predicts at grid points with

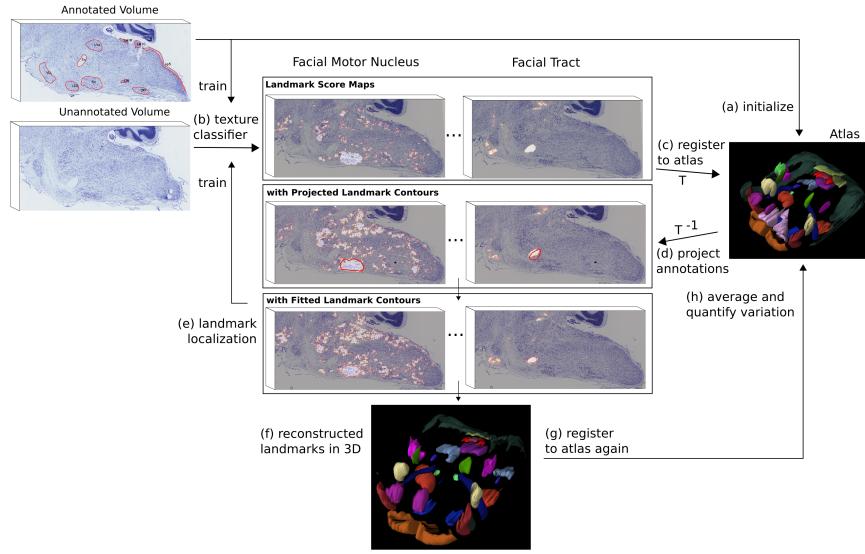


Figure 1. System overview. First, all image stacks are rigidly aligned. (a) One stack is annotated. The reconstructed 3D landmarks constitute an initial atlas. (b) A texture classifier is trained using patches extracted from annotated images. The classifier generates a set of score maps for each unannotated volume. (c) Unannotated volume is registered to the atlas based on the score maps using a 12-parameter affine transform. (d) Given the transform, the atlas projects landmark contours back onto the images. (e) Initialized by the projected contours, a snake finds the optimal contours based on the score maps. (f) The landmark contours are used to reconstruct the landmarks in 3D. (g) The volume is registered to the atlas again. (h) Statistics of landmarks' position and shape are updated in the atlas.

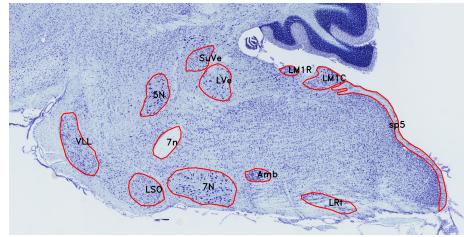


Figure 2. Human annotations on one section

a spacing of 56. The area between grid points is interpolated using bivariate spline. The result is a probability map for each image, where each pixel is assigned a probability vector.

3.2 Intra-stack Registration

As the brain sections are prepared using an advanced tape-transfer system, which almost perfectly preserves the shape of the sections, 3-parameter rigid transforms are sufficient to bring all sections into reasonable alignment. The transformations are found using the *elastix* program [7], based on maximizing the normalized correlation of the intensity between the thumbnail versions of the sections.

Once the registration transformations are known, a probability volume is reconstructed for each stack from the probability maps of all its sections. In a probability volume, each voxel has a probability vector. For the annotated stack, an annotation volume is reconstructed, in which each voxel has a scalar label. To facilitate registration at the next stage, the 3D contours of annotations are smoothed. We assume the sections all have constant thickness and are parallel to other sections in the same stack.

3.3 3D-3D Registration

To generate the atlas, we need to register the reconstructed volumes of all individuals in 3D and obtain an average. This process is done iteratively. We begin by using the annotated volume as the template and register the other volumes to it. Because the sections may be cut at different angles for different stacks, in effect resulting in shears between the volumes, we find 12-parameter 3D affine transformations.

Given the annotated volume Ω and a probability volume S , the best transformation is one that maximizes the “same-class overlap” objective:

$$M(T) = \sum_k \sum_{p \in \Omega_k} S_k[Tp] ,$$

where T is a 3-by-4 affine transform matrix, Ω_k is the set of voxels in the annotation volume that have label k , S_k is the volume formed by taking the entries corresponding to class k from the probability vectors of every voxel in S .

The optimization starts with grid search over the 3 translation parameters. Then gradient descent with Adagrad is used to approach the optimum.

3.4 Localizing Landmarks on Registered Images

To compute the statistics of landmarks’ position and shape, we need to accurately localize the landmarks of the registered volumes in 3D.

The fact that the probability volumes are registered to the annotation volume means the annotations can be propagated to these volumes. Figure 1 shows the contours of projected annotations on one of the registered images. These contours are not adapted to the actual images, but provide a rough guess for the landmarks’ positions. Starting

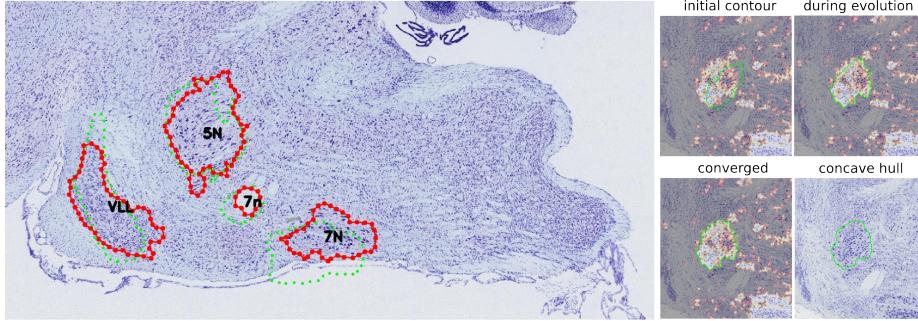


Figure 3. (a) Snake-localized contours (red), initialized by propagated contours from atlas (green). (b) The snake algorithm starts from the propagated contour. After convergence, a concave hull is computed to approximate it.

from them, a snake is used to fit the probability map of each landmark, giving more accurate contours (Figure 3).

A projected contour is first shifted to a position maximizing the average score inside. The snake algorithm then warps the contour to fit the score map of the particular landmark. The snake implementation used is the morphological ACWE [9]. After the snake converges, a concave hull is computed to simplify the contour.

Sometimes a landmark exists on an image, but the atlas does not project a contour on this image. So for a landmark without projected contour, we duplicate the projection if any of the neighboring 10 sections has it.

After a small amount of human editing, these contours essentially become annotations for these registered stacks. Patches can be extracted from them to improve the texture classifier. This automatic augmentation of the training data constitutes the semi-supervised scheme of iteratively learning the landmark detectors.

3.5 Generating the Atlas

The atlas includes the statistics of the position of each landmark.

Based on the transformations computed previously, landmark contours in registered images are reconstructed as 3D surface meshes in the common coordinate space defined by the annotation volume. The meshes are decimated and smoothed.

The principle axes of the 3D meshes are computed.

Amount of rigid transform that aligns a landmark from a new brain to the atlas.

4 Experiment Results

4.1 Variation of Landmark Position Between Brains

4.2 Accuracy of Classifier

The neural network is implemented using mxnet [2]. The first batch of training data consists of 1000 polygons. After landmarks are localized on images of each new stack,

the classifier is updated using the additional training data. The accuracy on the held-out brain improves with the number of brains registered to the atlas, as can be seen in Figure 1.

The confusion matrix (Figure 4) for patches extracted from a new brain shows the good generalizability of the texture classifier. There are several pairs/groups of landmarks that are often mistaken. As there is competition of probabilities among classes that are similar, resulting in undesirable suppression of the probability value of certain classes. A trick we used to remedy this is to identify similar classes and merge them into one class. They are usually not near each other, so we can do that.

The training accuracy quickly approaches 100%. The confusion matrix suggests

| | | Confusion matrix for test brain MDS94 | | | | | | | | | | | | | | | | | | | |
|------|------|---------------------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| | | True label | | | | | | | | | | | | | | | | | | | |
| | | Predicted label | | | | | | | | | | | | | | | | | | | |
| Back | 0.72 | 0.01 | 0.00 | 0.06 | 0.00 | 0.00 | 0.01 | 0.00 | 0.01 | 0.03 | 0.00 | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.02 | 0.01 | 0.00 | 0.02 | 0.01 |
| SA | 0.09 | 0.55 | 0.03 | 0.13 | 0.00 | 0.00 | 0.01 | 0.00 | 0.02 | 0.02 | 0.00 | 0.01 | 0.11 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 | 0.02 | 0.00 | 0.07 |
| Th | 0.12 | 0.05 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.07 |
| M | 0.03 | 0.05 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 |
| LN | 0.11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 |
| Cr | 0.25 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.05 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.14 | 0.00 | 0.00 | 0.17 |
| IVC | 0.17 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.17 | 0.00 | 0.03 | 0.00 | 0.00 | 0.03 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 |
| Pr | 0.17 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.00 | 0.01 | 0.11 | 0.00 | 0.00 | 0.00 | 0.07 | 0.01 | 0.05 | 0.00 | 0.02 |
| SuP | 0.38 | 0.06 | 0.00 | 0.06 | 0.00 | 0.00 | 0.01 | 0.00 | 0.24 | 0.11 | 0.01 | 0.00 | 0.04 | 0.00 | 0.00 | 0.00 | 0.02 | 0.03 | 0.00 | 0.04 | 0.00 |
| VLL | 0.12 | 0.01 | 0.00 | 0.02 | 0.00 | 0.00 | 0.01 | 0.02 | 0.72 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.05 |
| Or | 0.01 | 0.00 | 0.00 | 0.40 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.49 | 0.05 | 0.00 | 0.00 | 0.00 | 0.03 | 0.01 | 0.00 | 0.00 | 0.01 | 0.00 |
| Amb | 0.01 | 0.00 | 0.00 | 0.05 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 |
| H | 0.06 | 0.00 | 0.00 | 0.07 | 0.00 | 0.00 | 0.03 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 |
| Tz | 0.06 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.04 |
| Sol | 0.36 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.12 | 0.00 | 0.00 |
| Atg | 0.28 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.02 | 0.02 | 0.00 | 0.03 | 0.00 | 0.00 | 0.35 | 0.12 | 0.05 | 0.00 | 0.06 | 0.00 | 0.00 | 0.00 |
| LH | 0.01 | 0.00 | 0.00 | 0.19 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.76 | 0.00 | 0.00 | 0.02 |
| LC | 0.06 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Ar | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| qS | 0.37 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 | 0.63 |

Figure 4. Confusion matrix of the classifier

4.3 Accuracy of 3D Registration

4.4 Accuracy of Snake-localized Contours

We analyze the number of edits a human give on our snake-localized contour. Out of 100 contours, 20 are edited, which are of three types:

- 30% is moving vertices. This includes deleting a detection and add a new one at a different location. Among these, on average 40% of the vertices are moved, and the mean displacement is 5 um.
- 40% is removing spurious detection.
- 50% is adding annotations for missed landmarks. Failures are mostly due to the score maps given by the classifier is not optimal.

5 Conclusion

Future work includes:

- Our current atlas only characterized the position. For the shape we only characterized the rough orientation by treating all landmarks as an ellipsoid. This does not capture the shape of curved structures. The next step is to learn better 3D shape models.
-

References

1. Technical white paper: Allen mouse common coordinate framework. Tech. rep. (2015) [2](#)
2. Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., Zhang, Z.: Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. arXiv preprint arXiv:1512.01274 (2015) [4.2](#)
3. Chiang, A.S., Lin, C.Y., Chuang, C.C., Chang, H.M., Hsieh, C.H., Yeh, C.W., Shih, C.T., Wu, J.J., Wang, G.T., Chen, Y.C., et al.: Three-dimensional reconstruction of brain-wide wiring networks in drosophila at single-cell resolution. Current Biology 21(1), 1–11 (2011) [2](#)
4. Dong, H.: Allen Reference Atlas: a digital color brain atlas of the C57BL/6J male mouse. John Wiley & Sons (2008) [1, 2](#)
5. Gower, J., et al.: Generalized procrustes analysis. Psychometrika 40(1), 33–51 (1975)
6. Johnson, G.A., Badea, A., Brandenburg, J., Cofer, G., Fubara, B., Liu, S., Nissanov, J.: Waxholm space: an image-based reference for coordinating mouse brain research. Neuroimage 53(2), 365–372 (2010) [1, 2](#)
7. Klein, S., Staring, M., Murphy, K., Viergever, M.A., Pluim, J.P.: Elastix: a toolbox for intensity-based medical image registration. Medical Imaging, IEEE Transactions on 29(1), 196–205 (2010) [3.2](#)
8. Lein, E.S., Hawrylycz, M.J., Ao, N., Ayres, M., Bensinger, A., Bernard, A., Boe, A.F., Boguski, M.S., Brockway, K.S., Byrnes, E.J., et al.: Genome-wide atlas of gene expression in the adult mouse brain. Nature 445(7124), 168–176 (2007) [2](#)
9. Marquez-Neila, P., Baumela, L., Alvarez, L.: A morphological approach to curvature-based evolution of curves and surfaces. Pattern Analysis and Machine Intelligence, IEEE Transactions on 36(1), 2–17 (2014) [3.4](#)
10. Ng, L., Bernard, A., Lau, C., Overly, C.C., Dong, H.W., Kuan, C., Pathak, S., Sunkin, S.M., Dang, C., Bohland, J.W., et al.: An anatomic gene expression atlas of the adult mouse brain. Nature neuroscience 12(3), 356–362 (2009) [2](#)
11. Oh, S.W., Harris, J.A., Ng, L., Winslow, B., Cain, N., Mihalas, S., Wang, Q., Lau, C., Kuan, L., Henry, A.M., et al.: A mesoscale connectome of the mouse brain. Nature 508(7495), 207–214 (2014) [2](#)
12. Paxinos, G., Franklin, K.B.: The mouse brain in stereotaxic coordinates. Gulf Professional Publishing (2004) [1](#)
13. Peng, H., Chung, P., Long, F., Qu, L., Jenett, A., Seeds, A.M., Myers, E.W., Simpson, J.H.: Brainaligner: 3d registration atlases of drosophila brains. Nature methods 8(6), 493–498 (2011) [2](#)
14. Randler, O., Wee, C.L., Naumann, E.A., Nnaemeka, O., Schoppik, D., Fitzgerald, J.E., Portugues, R., Lacoste, A.M., Riegler, C., Engert, F., et al.: Whole-brain activity mapping onto a zebrafish brain atlas. Nature methods 12(11), 1039–1046 (2015) [2](#)
15. Ronneberger, O., Liu, K., Rath, M., Rueβ, D., Mueller, T., Skibbe, H., Drayer, B., Schmidt, T., Filippi, A., Nitschke, R., et al.: Vibe-z: a framework for 3d virtual colocalization analysis in zebrafish larval brains. Nature Methods 9(7), 735–742 (2012) [2](#)